2-stage stochastic linear programming

Yesterday, we considered problems that were modelling an extreme form of uncertainty. The decision maker, just knows the sort of problem he is going to be solved, a scheduling problem, a rent-or-buy problem, etc, but he does not know anything about the parameters of the problem.

Today we treat a model in which parameters are uncertain, but they can be modelled as stochastic variables. Let us first study a very general class of such stochastic optimisation problems.

Take a linear programming problem:

$$\begin{array}{rl} \max & cx \\ \text{subject to} & Ax \leq b \\ & \tilde{T}x \leq \tilde{\xi} \end{array}$$

with $b \in \mathbb{R}^m$, $\tilde{\xi} \in \mathbb{R}^d$, and A an $m \times n$ matrix, $\tilde{T}$ an $d \times n$ matrix, and $p \in \mathbb{R}^n$. The parameters with a tilde over them are stochastic variables. We know a probability distribution of their values, but we don't know their values exactly.

Clearly this problem is ill-defined, since a solution x that is optimal for one realisation of $\tilde{T}$ and $\tilde{\xi}$ may even be infeasible for another.

Two main directions have been taken in the literature to arrive at sensible models. In the conceptually easiest, violation of the uncertain constraints is allowed to occur with a probability that does not exceed a prespecified level, giving the so-called *probabilistic constraints problem.* The best comprehensive survey of this field is the book:

A. Prekopa, *Stochastic Programming*, Mathematics and its applications Vol. 324, Kluwer, Dordrecht, 1995.

We consider the other direction: the *2-stage stochastic programming problem* (or also called stochastic recourse problem). Conceptually one should think of the decision process taking place in two *stages.* In the first, values for the *first stage variables x* are chosen. In the second, upon a realisation of the random parameters, a *recourse* action is to be taken in case of infeasibilities. Costs are attached to the various possible recourse actions leading to the *second stage* (or recourse) problem, to choose the optimal action given the infeasibilities. The expected cost of the optimal recourse action is then added to the objective function. For a comprehensive review of the extensive literature we refer to

J. Birge, F. Louveaux, *Introduction to stochastic programming*, Springer-Verlag, New York, 1997.

A generic mathematical programming formulation for this problem is

$$\begin{array}{rl} \max & cx + \mathbb{E}[\max\{\tilde{q}y \mid Wy \le \tilde{T}x - \tilde{\xi},\ y \in \mathbb{R}^{n_1}\}] \\ \text{subject to} & Ax \le b. \end{array} \tag{1}$$

with $\tilde{q} \in \mathbb{R}^{n_1}$ and W an $d \times n_1$ matrix. In the literature W is sometimes allowed to be a random matrix. However, this may cause the feasible region to be non-convex in terms of x. We concentrate on the so-called *fixed recourse model* in which W is fixed. Moreover, we assume that W is such that for any x and any realisation of $\tilde{T}$ and $\tilde{\xi}$ there exists a feasible solution y in the second stage problem. This property of W is called the *complete recourse property*, and the model is accordingly called the *complete recourse model*.

The problem is often more compactly written as:

$$\begin{array}{rc} \max & cx + Q(x) \\ \text{subject to} & Ax \le b. \end{array}$$

with

$$Q(x) = \mathbb{E}[\max\{\tilde{q}y \mid Wy \le \tilde{T}x - \tilde{\xi},\ y \in \mathbb{R}^{n_1}\}]$$

Or we could even make it more compact by writing

$$Q(x) = \mathbb{E}[v(\tilde{T}x - \tilde{\xi})]$$

with

$$v(\tilde{T}x - \tilde{\xi}) = \max\{\tilde{q}y \mid Wy \le \tilde{T}x - \tilde{\xi},\ y \in \mathbb{R}^{n_1}\}.$$

Seen in this way $v(\tilde{T}x - \tilde{\xi})$ is the so-called value function of the second stage LP. It is well-known (and also not to hard to prove) that the value function of a maximisation problem is concave. Hence $Q(x)$ seen as a (possibly infinite) convex combination of value functions is also concave. Thus, in fact we have a concave optimisation problem with a polyhedral, hence convex, feasibility region. Hence, the theory of convex optimisation tells us that we can solve this by the ellipsoid method. However, if we think that this makes the 2-stage stochastic LP be in P then we overlook an essential complication as we will see soon.

Let us first study the modelling power of such a problem for combinatorial optimisation under uncertainty. After that we consider the complexity of solving stochastic linear programming problems and review some solution methods.

Modelling 2-stage combinatorial problems

Set Cover:
Given is a universe U of elements $e_1, \ldots, e_n$ and a family of sets $S_1, \ldots, S_m$, subsets of U. Set S_i has weight w_i. Find a collection of sets of total minimum weight such that each element is contained in (covered by) at least one of the sets in the collection.

This is a famous benchmark combinatorial optimisation problem. The greedy algorithm that always chooses as the next set the one that covers most yet uncovered elements has an approximation ratio of $\ln n$, and this is best possible for any polynomial time algorithm, essentially unless P=NP.

Now think of the following two-stage stochastic setting. There is uncertainty about the elements to be covered. Thus we know the universe but we don't know which elements will be required to be covered. In the stochastic programming setting each possible $A \subset U$ of elements has a probability p_A of occurring. Each such set is, what we call, a possible *scenario*.

We can decide to *buy* a set S_i in the first stage at first-stage (a priory) cost (weight) w_i^1 or in the second stage at second stage (a posteriory) cost w_i^2, or, of course, not buy S_i at all. I emphasize that in the first stage we buy sets without having seen the scenario that we will have to cover in the second stage. After having bought some sets in the first stage a subset $A \subset U$ is drawn according to the probabilities given, and then additional sets have to be bought in the second stage in order to cover the elements of A that have not been covered yet by the sets selected in the first stage.

The objective is to minimise the expected total weight: i.e., the sum of the total first stage cost and the expectation, over all second stage scenarios A, of the total second stage cost in case of scenario A.

We formulate it as a deterministic equivalent integer linear program (ILP): we use x_i to denote that S_i is selected in stage 1 ($x_i = 1$) or not ($x_i = 0$) and $r_{A,i}$ as the recourse variable that indicates if S_i is selected in the second stage under scenario A ($r_{A,i} = 1$) or not ($r_{A,i} = 0$). We write immediately the LP-relaxation.

$$\begin{array}{ll} \min & \sum_{i=1}^m w_i^1 x_i + \sum_{A \in 2^U} p_A \sum_{i=1}^m w_i^2 r_{A,i} \\ \text{s.t.} & \sum_{i:e \in S_i} x_i + \sum_{i:e \in S_i} r_{A,i} \geq 1 \ \forall A, \ \forall e \in A \\ & x_i, r_{A,i} \geq 0. \end{array}$$

Clearly a $\{0,1\}$ solution corresponds exactly to a feasible solution to our problem.

Suppose we have a solution (x, r) to the LP. For every element e we have $\sum_{i:e \in S_i} x_i \geq 1/2$ or in any scenario that contains e $\sum_{i:e \in S_i} r_{A,i} \geq 1/2$. Let

$E = \{e \mid \sum_{i:e\in S_i} x_i \geq 1/2\}$. Then $2x$ is a feasible (fractional) solution for the set cover problem restricted to E. Thus if we have an approximation algorithm for SET COVER that has approximation ratio ρ, then we can find a set cover of E of cost at most $\rho \sum_{i=1}^m w_i^1 x_i$. We take the sets in this set cover as our first stage decision $\bar{x}$.

Similarly, $2r_A$ is a feasible fractional set cover for all the elements in $A \setminus E$ and hence all these elements can be covered at cost bounded by $\rho \sum_{i=1}^m 2w_i^2 r_{A,i}$.

Thus, the first stage decision $\bar{x}$ gives a solution of cost at most 2ρ times the solution value of (x, r). In particular this implies that we obtain a solution that is within $2 \ln n$ from the optimal solution, under the assumption that we are able t find the optimal solution of the LP.

In the black box model the LP has an exponential number of variables and constraints. People that would hope that still some clever separation combined with column generation might work will be disappointed in the next part when I'll treat the complexity. Thus, it is no option to compute the optimal solution. On the other hand we only need to find the values for x, and indeed we have seen that we just need to find the set E from the LP and find an approximate set cover of E. So certainly the output does not need polynomial space.

Let us rewrite the problem in the compact form that I have presented before and then turn to solving 2-stage stochastic linear programming problems in general.

$$\begin{array}{ll} \min & \sum_{i=1}^m w_i^1 x_i + Q(x) \\ \text{s.t.} & x_i \geq 0. \end{array}$$

with

$$Q(x) = \sum_{A \in 2^U} p_A f_A(x)$$

and

$$f_A(x) = \begin{array}{c} \min \sum_{i=1}^m w_i^2 r_{A,i} \\ \text{s.t.} \sum_{i:e\in S_i} r_{A,i} \geq 1 - \sum_{i:e\in S_i} x_i \ \forall e \in A \\ r_{A,i} \geq 0. \end{array}$$

So our only hope is to avoid to make function evaluations of the expected second stage costs, or to find a way to calculate such function values efficiently. We will now see that the last option is hopeless unless the polynomial hierarchy collapses.

Exercise 2. Take your favourite combinatorial optimization problem and formulate a 2-stage stochastic programming version.

Complexity

Suppose the realisations of the stochastic variables come in *scenarios*: $(q^1, T^1, \xi^1), (q^2, T^2, \xi^2), \ldots, (q^K, T^K, \xi^K)$, with K denoting the total number of possible realisations. Each realization (q^k, T^k, ξ^k) has a probability p^k of occurrence. The problem can now be formulated as

$$\begin{array}{lllll} \max & cx + \sum_{k=1}^{K} p^k (q^k)^T y^k & & & \\ s.t. & Ax & & \leq b, & \\ & T^k x & + \quad W y^k & \leq \xi^k, & k = 1, \ldots, K. \end{array}$$

If, as input of the problem, each scenario and its corresponding probability has to be specified completely, then the input size of the problem is just the size of the binary encoding of all the parameters in this deterministic equivalent problem and hence the problem is polynomially solvable.

Another extreme is that the distribution is not specified at all. Samples can be requested for by an oracle. This is referred to as the *black box* model. It is not so difficult to prove that the objective function of (1) is *concave*. Therefore, the two-stage stochastic programming problem boils down to maximising a concave function over a convex (polyhedral) set, which seems to be doable, even in the black box model.

However, the complexity of the problem is highly dominated by any single evaluation of the objective function, which we show using a slightly less powerful model for the randomness: we assume that the parameters are independently distributed random variables. Under this model the problem is ♯P-hard. Since the proof is so easy, let me show it to you.

Consider the problem:

Definition 0.1. Graph reliability. Given a directed graph with m edges and n vertices, determine the *reliability* of the graph, defined as the probability that two given vertices u and v are connected, if each edge fails independently with probability 1/2.

This problem is ♯P-hard proven by L. Valiant, SIAM J. Comput. 1979.

♯P is a class of counting problems. Remember that NP is the class of decision problems asking if fr a given problem at least 1 solution exists. ♯P asks how many solutions exist. Thus, clearly NP⊂ ♯P. It is highly unlikely that NP= ♯P.

The reduction is now essentially giving a two-stage linear programming formulation for Graph reliability
Take any instance of graph reliability, i.e. a network $G = (V, A)$ with two

fixed vertices u and v in V. Introduce an extra edge from v to u, and introduce for each edge $(i,j) \in A$ a variable y_{ij}. Give each edge a random weight $\mathbf{q}_{ij}$ except for the edge (v,u) that gets a deterministic weight of 1. Let the weights be independent and identically distributed (i.i.d.) with distribution $\Pr\{\mathbf{q}_{ij} = -2\} = \Pr\{\mathbf{q}_{ij} = 0\} = 1/2$. The event $\{\mathbf{q}_{ij} = -2\}$ corresponds to failure of the edge (i,j) in the GRAPH RELIABILITY instance. If, for a realization of the failures of the edges, the network has a path from u to v, then there is a path from u to v consisting of edges with weight 0 only and vice versa.

Denote $A' = A \cup (v,u)$. Now define the two-stage stochastic programming problem:

$$\max\{-cx + Q(x) \mid 0 \leq x \leq 1\}$$

with

$$\begin{aligned} Q(x) = E[\max\{\sum_{(i,j)\in A} \mathbf{q}_{ij} y_{ij} + y_{vu} \mid \sum_{i:(i,j)\in A'} y_{ij} - \sum_{k:(j,k)\in A'} y_{jk} &= 0 \ \forall j \in V, \\ y_{ij} &\leq x \ \forall (i,j) \in A\}], \end{aligned}$$

where c is a parameter.

Suppose that for a realization of the failures of the edges there is a path from u to v in the network. As we argued the costs $q_{ij} = 0$ for edges (i,j) on the path. For such a realization, the optimal solution of the second-stage problem, is obtained by setting all y_{ij}'s corresponding to edges (i,j) on this path and y_{vu} equal to x, their maximum feasible value, and setting $y_{ij} = 0$ for all (i,j) not on the path. This yields solution value x for this realization.

Suppose that for a realization the graph does not have a path from u to v, implying in the reduced instance that on each path there is an edge with weight -2 and vice versa, then the optimal solution of the realized second-stage problem is obtained by setting all y_{ij}'s equal to 0, and also $y_{vu} = 0$, yielding solution value 0.

Therefore, the network has reliability R if and only if $Q(x) = Rx$. This implies immediately that evaluation of Q in a single point $x > 0$ is $\sharp$P-hard. Now it just remains to be shown that to solve a two-stage stochastic programming problem we need to have at least one evaluation of the objective function.

$Q(x) = Rx$ implies that the objective value of the 2-stage stochastic LP is $(R-c)x$. Hence, if $c < R$ then the optimal solution is $x = 1$ and the optimal value is $R - c$, whereas if $c \geq R$, the optimal solution is $x = 0$ with optimal value 0. Since R can have at most 2^m possible values, bisection search on c will reveal the correct value of R is we can solve the 2-stage stochastic LP.

What we proved is:

Theorem 0.1. *Two-stage stochastic programming with discrete distributions on the parameters is* $\sharp$P*-hard.*

In [Dyer & Stougie 2006] it is also shown that the same holds for continuous distributions. Also one can easily think of more than two stages. If the number of stages is part of the input of the problem then it is shown in [Dyer & Stougie 2006] that the problem is even PSPACE-hard.

Let me now sketch how 2-stage stochastic linear programming can be solved with a polynomial time randomised approximation scheme.

Methods for solving 2-stage stochastic LP

An efficient solution method for convex optimisation is the ellipsoid method by Khachyan. I will give a pictural explanation of this method on the blackboard. The proof of efficient theoretical performance is based on showing that the volume of the consecutive ellipsoids is shrinking in each step by a sufficient amount.

So far, nothing is different for 2-stage stochastic LP, under the assumption that we can find gradients or subgradients of the objective function efficiently. Just to remind you:

A *subgradient* of function g in the point x, restricted to a set P, is any vector d such that for any $y \in P$ we have $g(y) - g(x) \geq d(y - x)$.

It is clearly excluded that we can compute exact subgradients. Instead what we propose is to find an approximation (estimation)of the subgradient, called a ω-subgradient:

An ω*-subgradient* of function g in the point x, restricted to a set P, is any vector d such that for any $y \in P$ we have $g(y) - g(x) \geq d(y - x) - \omega g(x)$.

Clearly, we may not find the optimum, but get an approximate of the optimal solution with a value that differs from the optimal one by a factor depending on ω.

Let us see how we may find approximate subgradients for the LP-relaxation of the 2-stage stochastic set cover problem. Remember the objective function:

$$\min \quad \textstyle\sum_{i=1}^{m} w_i^1 x_i + Q(x)$$

with

$$Q(x) = \sum_{A \in 2^U} p_A f_A(x)$$

and

$$
\begin{aligned}
f_A(x) = \quad & \min \textstyle\sum_{i=1}^{m} w_i^2 r_{A,i} \\
& \text{s.t.} \textstyle\sum_{i:e\in S_i} r_{A,i} \geq 1 - \sum_{i:e\in S_i} x_i \ \forall e \in A \\
& r_{A,i} \geq 0.
\end{aligned}
$$

$\Delta Q(x) = \sum_{A\in 2^U} p_A \Delta f_A(x)$. Everyone knowing duality theory should know how to find $\Delta f_A(x)$. The dual of $f_A(x)$ is

$$
\begin{aligned}
f_A(x) = \quad & \max \textstyle\sum_e (1 - \sum_{i:e\in S_i} x_i) z_{A,e} \\
& \text{s.t.} \textstyle\sum_{e\in S_i} z_{A,e} \leq w_i^2 \ \forall S_i \\
& z_{A,e} = 0 \ \forall e \notin A.
\end{aligned}
$$

So, $-$ the optimal dual variables in scenario A give the gradient of $f_A(x)$. Hence $\Delta f(x)$ is obtained by taking the expectation over the all scenarios.

But expected values can be estimated by sampling! The rest is technical detail that I leave to yourself to read if you are interested.